AMENDMENTS

In the Claims:

Please amend the claims as indicated hereafter.

1. (Currently Amended) A computer system for processing instructions of a computer program, comprising:

a plurality of registers;

a plurality of connections corresponding respectively with said registers;

at least one pipeline configured to process and execute said instructions;

~~a scoreboard coupled to said plurality of connections and to said at least one pipeline, said scoreboard having a plurality of bits corresponding respectively with said plurality of registers, said scoreboard configured to transmit each of said bits across a different one of said connections, each of said bits indicative of whether a pending write to a corresponding one of said registers exists;~~

decoding circuitry coupled to said at least one pipeline, said decoding circuitry configured to decode ~~at least one~~ a plurality of encoded register identifiers associated with ~~at least one of~~ said instructions into a plurality of decoded register identifiers, [[said]] one of said decoded register identifiers having a plurality of bits ~~corresponding respectively with said plurality of registers and~~ identifying at least one of said registers, each bit of said one decoded register identifier corresponding with a respective one of said plurality of registers;

a scoreboard coupled to said plurality of connections and said decoding circuitry, said scoreboard having a plurality of bits corresponding respectively with said plurality of registers, said scoreboard configured to transmit each of said bits across a different one of said connections, each of said bits indicative of whether a pending write to a corresponding one of said registers exists, said scoreboard configured to update at least one of said bits of said scoreboard based on said one decoded register identifier; and

hazard detection circuitry coupled to each of said plurality of connections and to said decoding circuitry, said hazard detection circuitry configured to receive each of said decoded register identifiers from said decoding circuitry and to detect data hazards by comparing bits of ~~compare each of~~ said decoded register identifiers, including at least one bit of said one decoded register identifier, to said ~~transmitted~~ bits ~~to a respective one of said bits of said decoded register identifier and to detect data hazards based on comparisons of said transmitted bits to said decoded register identifier bits~~ transmitted by said scoreboard.

2. (Previously Presented) The system of claim 1, wherein said transmitted bits form a data word transmitted from said scoreboard to said hazard detection circuitry, each asserted bit in said data word indicating that a different one of said registers is associated with a pending write.

3. (Previously Presented) The system of claim 1, wherein said scoreboard includes a plurality of registers, each of said scoreboard registers containing a different one of said scoreboard bits and connected to a different one of said connections.

4-5. (Canceled).

6. (Currently Amended) The system of claim 1, wherein said one decoded register identifier includes a plurality of bits, wherein each of said bits in said decoded register identifier that corresponds to said at least one register is asserted, and wherein the remaining bits in said decoded register identifier are deasserted.

7. (Currently Amended) A system for processing instructions of computer programs, comprising:

at least one pipeline;

a plurality of registers;

a plurality of connections, each of said connections corresponding to a different one of said registers;

a decoder configured to decode a register identifier from said at least one pipeline to a decoded register identifier having a first plurality of bits, each of said first plurality of bits corresponding with a respective one of said plurality of registers, said decoded register identifier identifying at least one of said registers;

means for maintaining a second plurality of bits and for indicating via said second plurality of bits which of said registers is associated with a pending write, said maintaining means configured to transmit said second plurality of bits across said connections, wherein each bit transmitted across each of said connections is indicative of whether the register corresponding to said each connection is associated with a pending write, said maintaining means further configured to update said second plurality of bits based on said decoded register identifier; and

hazard detection circuitry configured to perform comparisons between said second plurality of bits and [[a]] said decoded register identifier ~~associated with at least one instruction presently in said at least one pipeline~~, said hazard detection circuitry further configured to detect at least one data hazard based on said comparisons.

8. (Currently Amended) The system of claim 7, wherein said maintaining means includes a plurality of registers, each of said registers of said maintaining means containing a different one of said second plurality of bits and connected to a different one of said connections.

9. (Canceled)

10. (Currently Amended) ~~The system of claim 9,~~ A system for processing instructions of computer programs, comprising:

at least one pipeline;

a plurality of registers;

a plurality of connections, each of said connections corresponding to a different one of said registers;

means for maintaining a plurality of bits and for indicating via said bits which of said registers is associated with a pending write, said maintaining means configured to transmit said bits across said connections, wherein each bit transmitted across each of said connections is indicative of whether the register corresponding to said each connection is associated with a pending write;

hazard detection circuitry configured to perform comparisons between said bits and a decoded register identifier associated with at least one instruction presently in said at least one pipeline, said hazard detection circuitry further configured to detect at least one data hazard based on said comparisons; and

means for decoding, into said decoded register identifier, an encoded register identifier associated with said at least one instruction, said decoding means configured to transmit said decoded register identifier to said hazard detection circuitry and to said maintaining means, said decoded register identifier identifying at least one of said registers,

wherein said decoded register identifier includes a plurality of bits, wherein each of said bits in said decoded register identifier that corresponds to said at least one register is asserted, and wherein the remaining of said bits in said decoded register identifier are deasserted.

11. (Currently Amended) A method for processing instructions of computer programs, comprising the steps of:

processing said instructions via at least one pipeline;

providing a plurality of registers;

receiving an encoded register identifier from said at least one pipeline;

decoding said received register identifier into a decoded register identifier;

updating at least one of said bits based on said decoded register identifier;

maintaining a plurality of bits, each of said bits indicating whether a corresponding one of said registers is associated with a pending write;

transmitting a data word, said data word including each of said bits, wherein each asserted bit in said data word indicates that a different one of said registers is associated with a pending write;

receiving said data word;

comparing said data word to [[a]] said decoded register identifier ~~associated with at least one instruction presently in said at least one pipeline~~; and

detecting a data hazard based on said comparing step.

12. (Canceled)

13. (Currently Amended) A method for processing instructions of computer programs, comprising the steps of:

processing said instructions via at least one pipeline;

providing a plurality of registers;

decoding at least one register identifier associated with at least one of said instructions into a decoded register identifier having a plurality of bits, each of said bits corresponding with a respective one of said plurality of registers, said decoded register identifier identifying at least one of said registers;

maintaining a plurality of bits within a scoreboard, each of said scoreboard bits respectively corresponding with one of said registers;

providing a plurality of connections, each of said connections respectively corresponding with one of said registers;

indicating, via said scoreboard bits, which of said registers are associated with pending writes;

transmitting, from said scoreboard, each of said scoreboard bits across a different one of said connections;

~~decoding at least one register identifier associated with at least one of said instructions into a decoded register identifier, said decoded register identifier having a plurality of bits corresponding respectively with said plurality of registers and identifying at least one of said registers;~~

updating at least one of said scoreboard bits in said scoreboard based on said decoded register identifier;

comparing [[each]] at least one of said transmitted bits to ~~a respective~~ at least one of said bits of said decoded register identifier; and

detecting a data hazard based on said comparing step.

14-15. (Canceled).

16. (Currently Amended) A system for processing instructions of computer programs, comprising:

a plurality of registers;

a plurality of connections;

at least one pipeline;

a decoder configured to receive an encoded register identifier from said at least one pipeline and to decode said received register identifier into a decoded register identifier identifying at least one of said registers;

a scoreboard having data indicative of whether each of said plurality of registers is respectively associated with a pending write, said scoreboard configured to transmit said data across said connections and to update said data based on said decoded register identifier; and

hazard detection circuitry coupled to said connections and configured to receive said transmitted data and to perform a comparison between said transmitted data and [[a]] said decoded register identifier ~~associated with an instruction presently in said at least one pipeline~~, said hazard detection circuitry further configured to detect a data hazard based on said comparison.

17. (Canceled)

18. (Currently Amended) The system of claim [[17]] 16, wherein said transmitted data comprises a plurality of bits, each of said bits indicative of whether a corresponding one of said registers is associated with a pending write.

19. (Currently Amended) A method for processing instructions of computer programs, comprising the steps of:

processing said instructions via at least one pipeline;

using a plurality of registers to execute said instructions;

receiving from said at least one pipeline an encoded register identifier identifying at least one of said registers;

decoding said received register identifier into a decoded register identifier identifying each of said register identified by said encoded register identifier;

storing, in a scoreboard, data indicative of which of said registers is associated with a pending write;

transmitting said data from said scoreboard;

decoding ~~an encoded~~ said received register identifier ~~associated with one of said instructions~~ into a decoded register identifier identifying each of said registers identified by said encoded register identifier;

updating said data in said scoreboard based on said decoded register identifier;

comparing said transmitted data to said decoded register identifier; and

detecting a data hazard based on said comparing step.

20. (Previously Presented) The method of claim 19, wherein said transmitted data comprises a plurality of bits, each of said bits indicative of whether a corresponding one of said registers is associated with a pending write.

21. (Canceled)

22. (New) The system of claim 7, wherein each of said first plurality of bits corresponding to said at least one register identified by said decoded register identifier is asserted, and wherein the remaining of said first plurality of bits are deasserted.

23. (New) The method of claim 11, wherein said decoded register identifier includes a plurality of bits identifying at least one of said registers, wherein each of said bits in said decoded register identifier corresponds with a respective one of said registers, wherein each of said bits in said decoded register identifier corresponding to said at least one register identified by said decoded register identifier is asserted, and wherein the remaining of said bits in said decoded register identifier are deasserted.

24. (New) The method of claim 13, wherein each of said bits of said decoded register identifier corresponding to said at least one register identified by said decoded register identifier is asserted, and wherein the remaining of said first plurality of bits are deasserted.

25. (New) The system of claim 16, wherein said decoded register identifier includes a plurality of bits, wherein each of said bits in said decoded register identifier corresponds with a respective one of said registers, wherein each of said bits in said decoded register identifier corresponding to said at least one register identified by said decoded register identifier is asserted, and wherein the remaining of said bits in said decoded register identifier are deasserted.

26. (New) The method of claim 19, wherein said decoded register identifier includes a plurality of bits, wherein each of said bits in said decoded register identifier corresponds with a respective one of said registers, wherein each of said bits in said decoded register identifier corresponding to said at least one register identified by said decoded register identifier is asserted, and wherein the remaining of said bits in said decoded register identifier are deasserted.